



CPSC203(DATABASES): WEEK-1 LAB-1

INTRO TO DATABASES (QUICK):

BASIC PARTS OF A DATABASE

-Prepared By
Nashad Ahmed Safa
Graduate Student
Department of Computer Science

COURSE WEBSITE

[http://wiki.ucalgary.ca/page/
Courses/Computer_Science/
CPSC_203/CPSC_203_Template](http://wiki.ucalgary.ca/page/Courses/Computer_Science/CPSC_203/CPSC_203_Template)



BACKGROUND MATERIAL RESOURCES

- Computer Science Illuminated, Third Edition by Nell Dale and John Lewis
ISBN-13:978-0-7637-4149-5
- <http://www.jbpub.com>
- Lab 8 page 177 of Fluency Computer Skills Workbook



INTRODUCTION TO DATABASES

Today's tutorial introduces the basic parts of a database, using MS Access.

- At some point in our lives we all have (contact lists, records, finances, etc.) A spreadsheet is a good starting point, but often when there's a lot of information -a database is more efficient.



BASIC FOUNDATIONS OF DATABASES

Elements of databases

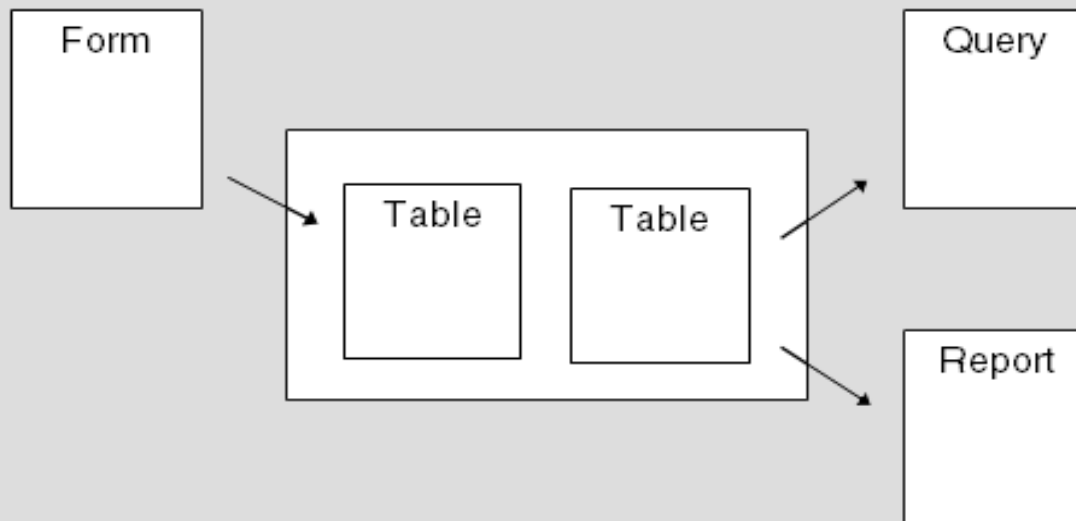
A database stores information in an organized way, and makes it easy to get information in and out.

Tables store data within the database.

Forms make it easy to put data into tables.

Queries pull out specific data.

Reports put data in an easily-read format.




BASIC FOUNDATIONS OF DATABASES

A database is defined as a structured set of data. We're dealing with a particular type of database called a Relational Database.

'Relational Databases' are based on Set Theory in Mathematics.

A database management system (DBMS) is a mixture of software and data that consist of:


1. The physical collection of files that contain data.
 2. The software that allows users to interact with the database and make modifications.
 3. The schema that specifies the logical structure of how the data is to be stored.
- 

BASIC FOUNDATIONS OF DATABASES

A database is defined as a structured set of data. We're dealing with a particular type of database called a Relational Database.

'Relational Databases' are based on Set Theory in Mathematics.

A database management system (DBMS) is a mixture of software and data that consist of:

1. The physical collection of files that contain data.
 2. The software that allows users to interact with the database and make modifications.
 3. The schema that specifies the logical structure of how the data is to be stored.
- 

BASIC FOUNDATIONS OF DATABASES

The type of database model we will be examining is the **relational model**, which is database model that organizes data and the relationship among them into tables. A **table** is defined as a collection of **records**. A record is a collection of **related fields**. Each field of a database table represents a single piece of data that is stored.

- As an example, consider the database table in the next slide. It contains information about people (presumably an address book). Each row in the table corresponds to a record.



BASIC FOUNDATIONS OF DATABASES

Each record in the table is made up of the same fields in which a specific piece of data is stored. This is similar to how data is organized in spreadsheets.

<u>PersonID</u>	<u>Firstname</u>	<u>Lastname</u>	email	birthday
101	Rick	<u>Edwards</u>	rick.edwards@email.com	7-Mar-68
102	Jimmy	Foster	jimmy.foster@email.com	28-Feb-87
103	Nathan	<u>Garcia</u>	nathan.garcia@email.com	2-Jun-82
104	Louise	Knight	louise.knight@email.com	12-Dec-67
105	Gary	<u>Knox</u>	gary.knox@email.com	1-Dec-92
106	Rafael	Lorenz	rafael.lorenz@email.com	9-Jul-78
107	Veronica	Page	veronica.page@email.com	9-Sep-45
108	Hector	<u>Sanchez</u>	hector.sanchez@email.com	1-Apr-00
109	Billy	Smith	billy.smith@email.com	30-Aug-99
110	<u>Ricardo</u>	<u>Stuckey</u>	ricardo.stuckey@email.com	17-Nov-55
111	Ken	Weaver	ken.weaver@email.com	13-May-45
112	<u>Lorenzo</u>	West	lorenzo.west@email.com	1-Jun-84

BASIC FOUNDATIONS OF DATABASES


- Each record in the address book has a personID, first name, last name, email, and birthday field that contains the specific data.
- Usually one or more fields of a table have a key field. This field uniquely identifies a record in a database among other records in a table. In the address book table example, the personID field is the logical choice. That way if two or more people have the same first and last name they can still be uniquely identified and separated.
- The structure of the table corresponds to the schema that it represents.

SCHEMA OF A DATABASE

- The schema is an expression of the attributes of the records in a table, we can represent the schema of the address book as follows:

Address Book (PersonID:key, Fir stname, Lastname, Email, Birthday)

The advantage of using a relational database to store this type of information instead of a spreadsheet is the ability of a relational database management system to create tables that link various tables together.



RELATIONSHIPS

- **Records** represent a collection of fields stored in a table. We can create records that combine or exclude fields to gather all the information we require.
- With our address book example, we will add on two fields to the database schema. We're going to add City and Province to the database schema. Because these two fields have information that repeats themselves, there is no point in creating this type of information repeatedly. For example, if one or more people live in the city of "Calgary" it would be easier to store that information in a separate table and then reference the city using a unique number ID.
- In a relational database, the goal is to **avoid**

RELATIONSHIPS-EXAMPLE(1)

<u>PersonID</u>	<u>Firstname</u>	<u>Lastname</u>	<u>email</u>	<u>birthday</u>	<u>City</u>	<u>Province</u>
101	Rick	<u>Edwards</u>	rick.edwards@email.com	7-Mar-68	Calgary	Alberta
102	Jimmy	Foster	jimmy.foster@email.com	28-Feb-87	Vancouver	British Columbia
103	Nathan	<u>Garcia</u>	nathan.garcia@email.com	2-Jun-82	Edmonton	Alberta
104	Louise	Knight	louise.knight@email.com	12-Dec-67	Ottawa	Ontario
105	Gary	<u>Knox</u>	gary.knox@email.com	1-Dec-92	Regina	Saskatchewan
106	Rafael	Lorenz	rafael.lorenz@email.com	9-Jul-78	Surrey	British Columbia
107	Veronica	Page	veronica.page@email.com	9-Sep-45	Richmond	British Columbia
108	Hector	<u>Sanchez</u>	hector.sanchez@email.com	1-Apr-00	<u>Markham</u>	Ontario
109	Billy	Smith	billy.smith@email.com	30-Aug-99	Winnipeg	Manitoba
110	<u>Ricardo</u>	<u>Stuckey</u>	ricardo.stuckey@email.com	17-Nov-55	Toronto	Ontario
111	Ken	Weaver	ken.weaver@email.com	13-May-45	Hamilton	Ontario
112	<u>Lorenzo</u>	West	lorenzo.west@email.com	1-Jun-84	Montreal	Quebec

Same data will be used multiple times-not an efficient schema. We can use relationship to make the design more efficient



RELATIONSHIPS-EXAMPLE(2)

PersonID	Firstname	Lastname	email	birthday	CityID	ProvinceID
101	Rick	Edwards	rick.edwards@email.com	7-Mar-68	1	2
102	Jimmy	Foster	jimmy.foster@email.com	28-Feb-87	3	1
103	Nathan	Garcia	nathan.garcia@email.com	2-Jun-82	9	2
104	Louise	Knight	louise.knight@email.com	12-Dec-67	2	5
105	Gary	Knox	gary.knox@email.com	1-Dec-92	6	3
106	Rafael	Lorenz	rafael.lorenz@email.com	9-Jul-78	5	1
107	Veronica	Page	veronica.page@email.com	9-Sep-45	4	1
108	Hector	Sanchez	hector.sanchez@email.com	1-Apr-00	7	5
109	Billy	Smith	billy.smith@email.com	30-Aug-99	11	4
110	Ricardo	Stuckey	ricardo.stuckey@email.com	17-Nov-55	10	5
111	Ken	Weaver	ken.weaver@email.com	13-May-45	8	5
112	Lorenzo	West	lorenzo.west@email.com	1-Jun-84	12	6


CityID	City
1	Calgary
2	Ottawa
3	Vancouver
4	Richmond
5	Surrey
6	Regina
7	Markham
8	Hamilton
9	Edmonton
10	Toronto
11	Winnipeg

ProvinceID	Province
1	British Columbia
2	Alberta
3	Saskatchewan
4	Manitoba
5	Ontario
6	Quebec


This design uses the relationship aspect to avoid using duplicate data. Here the cities and provinces are created once and then referenced multiple times. Also the table is smaller as it is only having to track numbers instead of containing the entire name of a city or province. That information can be referenced and pulled as needed when the user requires the city and/or province. Also if the user does not require the city or province then it is ignored and the results can be retrieved faster.

AN EXAMPLE RELATIONAL DATABASE OF A COMPANY

EMPLOYEE						
SIN	Fname	Lname	Salary	Number	City	Dnumber
17171714	Debra	Beacon	70000	15	Calgary	1
1588752	Sam	Field	40000	15	Calgary	1
125485	Rajeet	Folk	78000	123	Calgary	2
25485520	Marie	Band	53500	2828	Torronto	2
6548525	Saleh	Dice	90400	66	Torronto	3



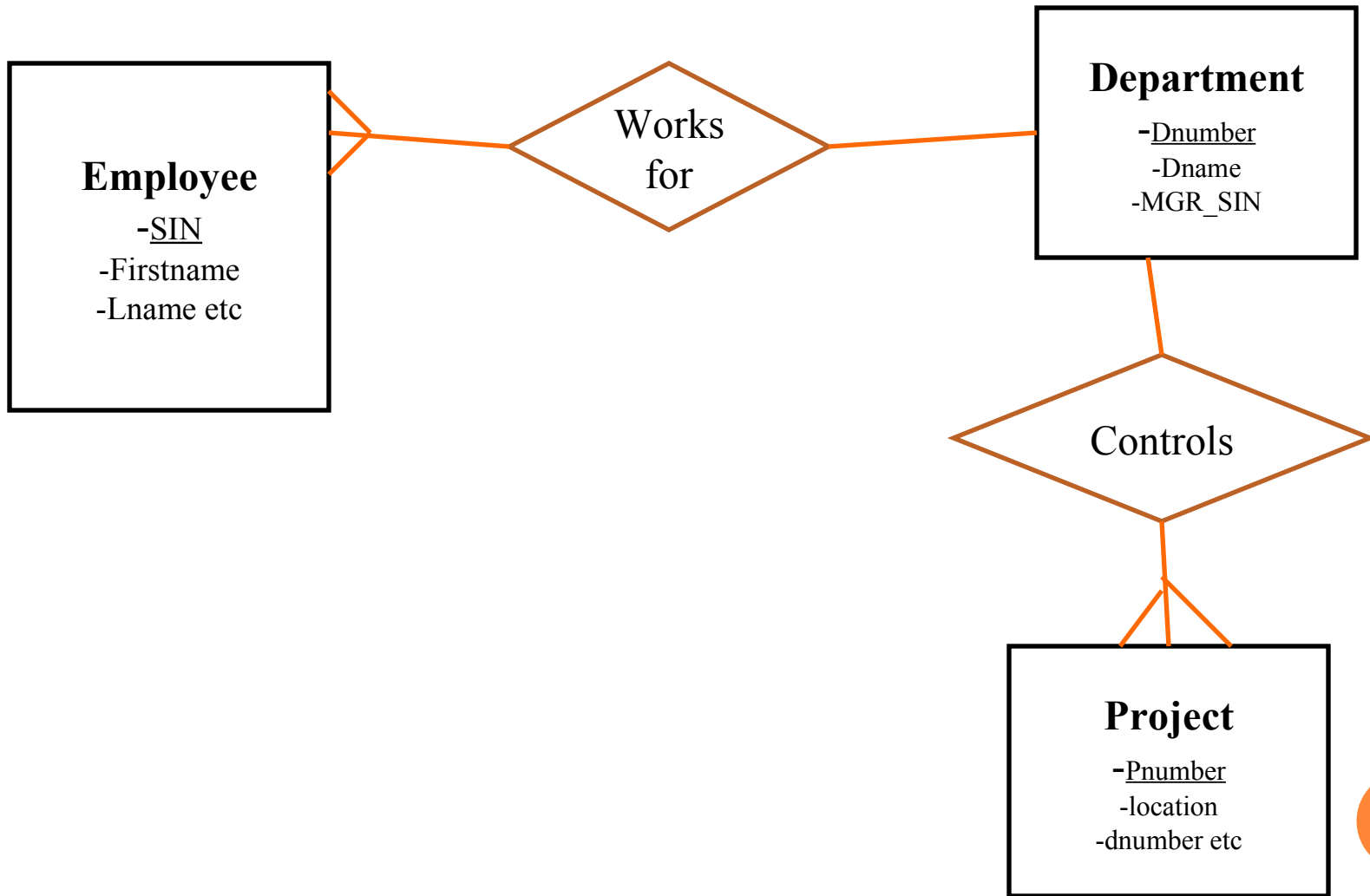
DEPARTMENT		
Dnumber	Dname	MGR_SIN
1	IT	17171714
2	Finance	125485
3	Marketing	6548525



PROJECT			
Pnumber	Pname	Location	Dnumber
1	Web shopping	Calgary	1
2	Network Upgrade	Calgary	1
3	New Benefits	Calgary	2
4	Product XT345	Torronto	3



ENTITY RELATIONSHIP DIAGRAM(ERD)



A SIMPLE EXERCISE

- Try to Design a Relational Database Schema and Corresponding ERD for a simple musical library having the following attributes:

Singer Name

Singer Biography

Singer Website

Album Title

Album Release Date

Song Title

Song Duration

Song Genre

